

## CLAIMS:

1. A micro-controller configuration apparatus operable to identify a set of register values for storage within registers of a micro-controller which includes a plurality of peripherals, the micro-controller being operable to route different selections of signals to and from said peripherals via a set of ports on the basis of the register values stored in said registers, said configuration apparatus comprising:

a first user input interface operable to enable a user to input data identifying a selection of peripherals from the plurality of peripherals included in said micro-controller;

a determination module operable to determine for input data identifying a selection of peripherals, a set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function, said determination module further being operable to identify a plurality of combinations of register values which when stored within registers of a micro-controller enable all of the determined signals to be routed via ports of said micro-controller;

a second user input interface operable to enable a user to select one of said plurality of combinations of register values identified by said determination module; and

an output module operable to output data identifying a set of register values for storage within registers in a micro-controller including register values corresponding to a selected combination of register values.

2. An apparatus in accordance with claim 1 wherein said determination module comprises:

a plurality of configuration tables each identifying for a port of a micro-controller, signals to be received or output via one or more pins associated with said port when a register associated with said port is set to a defined value;

a peripheral database configured to store data identifying for each of said plurality of peripherals, the signals required for routing to and from said selected peripherals to enable each of said peripherals to function; and

a configuration checking module operable to utilise said configuration tables and said peripheral database to identify combinations of register values which when stored within registers of a micro-controller enable all of signals required by an identified selection of peripherals to be routed via said ports of said micro-controller.

3. An apparatus in accordance with claim 1 wherein said determination module is operable to determine whether any combinations of register values which when stored

within registers of a micro-controller enable all of signals required by an identified selection of peripherals to be routed via said ports of said micro-controller and to output warning data if no combination of register values enables all of the signals required by an identified selection of peripherals to be routed via said ports of said micro-controller.

5

4. An apparatus in accordance with claim 3 wherein said warning data comprises data identifying which peripherals of an identified selection of peripherals require incompatible register values to be stored within registers of a micro-controller to enable all of the signals required by said identified selection of peripherals to be routed via said ports of said micro-controller.

10

5. An apparatus in accordance with claim 3 wherein said warning data comprises data identifying an alternative micro-controller including said identified set of peripherals and which is capable of storing a set of register values which enable all of the signals required by said identified selection of peripherals to be routed via said ports of said alternative micro-controller.

15

6. An apparatus in accordance with claim 1 wherein said second user input interface is operable to generate a schematic representation of said micro-controller illustrating the relative physical locations of pins of the micro-controller utilised to route signals to and from an identified selection of peripherals when register values including register values corresponding to said combinations of register values are stored within the registers of said micro-controller.

20

7. An apparatus in accordance with claim 6 wherein said second user input interface is operable to order said plurality of combinations of register values and generate a said schematic representation on the basis of a user selection of a combination from said ordered selection of combinations.

25

8. An apparatus in accordance with claim 7 wherein said second user input interface is operable to order said plurality of combinations of register values on the basis of the number of said pins of said micro-controller dedicated to routing signals for general purpose inputs and outputs when said combinations of register values are stored within the registers of said micro-controller.

30

35

9. An apparatus in accordance with claim 7 wherein said second user input interface is operable to order said plurality of combinations of register values on the basis of the relative distances between representations of said selected peripherals within said display and said representations of pins associated with routing signals to said respective peripherals when  
5 said combinations of register values are stored within the registers of said micro-controller.

10. An apparatus in accordance with claim 1 wherein said micro-controller further comprises one or more registers operable to store settings data for said plurality of peripherals, said apparatus further comprising a third user input interface operable to  
10 enable a user to input data identifying register values for storage in said one or more registers associated with said selection of peripherals.

11. An apparatus in accordance with claim 1 wherein said output module is operable to output data identifying a set of register values comprising register values corresponding to a  
15 selected combination of register values and default values for registers for which no other register values are identified.

12. An apparatus in accordance with claim 1 wherein said output module is operable to generate a computer program operable when stored in a memory associated with a said  
20 micro-controller to cause said micro-controller to store register values corresponding to said identified set of register values in registers in said micro-controller.

13. An apparatus in accordance with claim 1 wherein said output module is operable to output data defining a schematic representation of a said micro-controller including labels  
25 identifying said set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function.

14. An apparatus in accordance with claim 13 wherein said output module is operable to output data defining a schematic representation wherein said labels are grouped in said  
30 representation by the peripherals associated with said identified signals.

15. An apparatus in accordance with claim 13 wherein said output module is operable to output data defining a schematic representation wherein said labels are grouped in said  
representation so as to represent the physical locations of pins transferring said signals.

16. An apparatus in accordance with claim 1 wherein at least one of said ports of said micro-controller comprises a port operable to transfer a single signal selected on the basis of a stored register value in a register associated with said port.

5 17. An apparatus in accordance with claim 16 wherein each of said ports of said micro-controller comprises a port operable to transfer a single signal selected on the basis of a stored register value in a register associated with said port, said apparatus being operable to identify sets of register values which when stored in said registers enables the signals required for an identified selection of peripherals to function to be transferred via said ports.

10

18. A method of identifying a set of register values for storage within registers of a micro-controller which includes a plurality of peripherals, the micro-controller being operable to route different selections of signals to and from said peripherals via a set of ports on the basis of the register values stored in said registers, said method comprising:

15 receiving data identifying a selection of peripherals from the plurality of peripherals included in said micro-controller;

determining for received data identifying a selection of peripherals, a set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function;

20 identifying a plurality of combinations of register values which when stored within registers of a micro-controller enable all of the determined signals to be routed via ports of said micro-controller;

receiving data identifying a selection of one of said plurality of identified combinations of register values; and

25 outputting data identifying a set of register values for storage within registers in a micro-controller including register values corresponding to a selected combination of register values.

19. A method in accordance with claim 18 further comprising:

30 storing a plurality of configuration tables each identifying for a port of a micro-controller, signals to be received or output via one or more pins associated with said port when a register associated with said port is set to a defined value;

storing a peripheral database configured to store data identifying for each of said plurality of peripherals, the signals required for routing to and from said selected peripherals  
35 to enable each of said peripherals to function; and

utilising said configuration tables and said peripheral database to identify combinations of register values which when stored within registers of a micro-controller enable all of signals required by an identified selection of peripherals to be routed via said ports of said micro-controller.

5

20. A method in accordance with claim 18 further comprising:

outputting data identifying which peripherals of an identified selection of peripherals require incompatible register values to be stored within registers of a micro-controller to enable all of the signals required by said identified selection of peripherals to be routed via said ports of said micro-controller if no compatible set of register values can be identified.

10

21. A method in accordance with claim 18 further comprising:

outputting data identifying an alternative micro-controller including said identified set of peripherals, which is capable of storing a set of register values which enable all of the signals required by said identified selection of peripherals to be routed via said ports of said alternative micro-controller if no compatible set of register values can be identified.

15

22. A method in accordance with claim 18 further comprising:

generating data defining a schematic representation of said micro-controller illustrating the relative physical locations of pins of the micro-controller utilised to route signals to and from an identified selection of peripherals when register values including register values corresponding to said combinations of register values are stored within the registers of said micro-controller.

20

23. A method in accordance with claim 22 further comprising ordering said identified of combinations of register values; and

generating said schematic representation on the basis of a received user input selection of a combination from said ordered selection of combinations.

25

24. A method in accordance with claim 23 wherein said ordering of said plurality of combinations of register values comprises ordering on the basis of the number of said pins of said micro-controller dedicated to routing signals for general purpose inputs and outputs when said combinations of register values are stored within the registers of said micro-controller.

30

35

25. A method in accordance with claim 23 wherein said ordering of said plurality of combinations of register values comprises ordering on the basis of the relative distances between representations of said selected peripherals within a representative of said micro-controller and representations of pins associated with routing signals to said respective peripherals when said combinations of register values are stored within the registers of said micro-controller.

26. A method in accordance with claim 18 wherein said micro-controller further comprises one or more registers operable to store settings data for said plurality of peripherals, said method further comprising receiving data identifying register values for storage in said one or more registers associated with said selection of peripherals.

27. A method in accordance with claim 18 comprising outputting data identifying a set of register values comprising register values corresponding to the selected combination of register values and default values for registers for which no other register values are identified.

28. A method in accordance with claim 18 further comprising generating a computer program operable when stored in a memory associated with a said micro-controller to cause said micro-controller to store register values corresponding to said identified set of register values in registers in said micro-controller.

29. A method in accordance with claim 18 further comprising outputting data defining a schematic representation of a said micro-controller including labels identifying said set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function.

30. A method in accordance with claim 29 wherein said labels are grouped in said representation by the peripherals associated with said identified signals.

31. A method in accordance with claim 29 wherein said labels are grouped in said representation so as to represent the physical locations of pins transferring said signals.

32. A method in accordance with claim 18 wherein at least one of said ports of said micro-controller comprises a port operable to transfer a single signal selected on the basis of a stored register value in a register associated with said port.

5 33. A method in accordance with claim 32 wherein each of said ports of said micro-controller comprises a port operable to transfer a single signal selected on the basis of a stored register value in a register associated with said port, said method comprising identifying sets of register values which when stored in said registers enables the signals required for an identified selection of peripherals to function to be transferred via said ports.

10 34. A computer readable medium storing computer implementable instructions to cause a programmable computer to perform a method for identifying a set of register values for storage within registers of a micro-controller which includes a plurality of peripherals, the micro-controller being operable to route different selections of signals to and from said peripherals via a set of ports on the basis of the register values stored in said registers, the method comprising:

receiving data identifying a selection of peripherals from the plurality of peripherals included in said micro-controller;

15 determining for received data identifying a selection of peripherals, a set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function;

identifying a plurality of combinations of register values which when stored within registers of a micro-controller enable all of the determined signals to be routed via ports of said micro-controller;

25 receiving data identifying a selection of one of said plurality of identified combinations of register values; and

outputting data identifying a set of register values for storage within registers in a micro-controller including register values corresponding to a selected combination of register values.

30 35. A computer readable medium in accordance with claim 34 wherein said method further comprises:

storing a plurality of configuration tables each identifying for a port of a micro-controller, signals to be received or output via one or more pins associated with said port  
35 when a register associated with said port is set to a defined value;

storing a peripheral database configured to store data identifying for each of said plurality of peripherals, the signals required for routing to and from said selected peripherals to enable each of said peripherals to function; and

5 utilising said configuration tables and said peripheral database to identify combinations of register values which when stored within registers of a micro-controller enable all of signals required by an identified selection of peripherals to be routed via said ports of said micro-controller.

10 36. A computer readable medium accordance with claim 34 wherein said method further comprises:

outputting data identifying which peripherals of an identified selection of peripherals require incompatible register values to be stored within registers of a micro-controller to enable all of the signals required by said identified selection of peripherals to be routed via said ports of said micro-controller if no compatible set of register values can be identified.

15

37. A computer readable medium in accordance with claim 34 wherein said method further comprises:

20 outputting data identifying an alternative micro-controller including said identified set of peripherals, which is capable of storing a set of register values which enable all of the signals required by said identified selection of peripherals to be routed via said ports of said alternative micro-controller if no compatible set of register values can be identified.

38. A computer readable medium in accordance with claim 34 further comprising:

25 generating data defining a schematic representation of said micro-controller illustrating the relative physical locations of pins of the micro-controller utilised to route signals to and from an identified selection of peripherals when register values including register values corresponding to said combinations of register values are stored within the registers of said micro-controller.

30 39. A computer readable medium in accordance with claim 38 further comprising:

ordering said identified of combinations of register values; and  
generating said schematic representation on the basis of a received user input selection of a combination from said ordered selection of combinations.



40. A computer readable medium in accordance with claim 39 wherein said ordering of said plurality of combinations of register values comprises ordering on the basis of the number of said pins of said micro-controller dedicated to routing signals for general purpose inputs and outputs when said combinations of register values are stored within the registers of said micro-controller.

41. A computer readable medium in accordance with claim 39 wherein said ordering of said plurality of combinations of register values comprises ordering on the basis of the relative distances between representations of said selected peripherals within a representative of said micro-controller and representations of pins associated with routing signals to said respective peripherals when said combinations of register values are stored within the registers of said micro-controller.

42. A computer readable medium in accordance with claim 34 wherein said micro-controller further comprises one or more registers operable to store settings data for said plurality of peripherals, the method further comprising receiving data identifying register values for storage in said one or more registers associated with said selection of peripherals.

43. A computer readable medium in accordance with claim 34 wherein the method further comprises outputting data identifying a set of register values comprising register values corresponding to the selected combination of register values and default values for registers for which no other register values are identified.

44. A computer readable medium in accordance with claim 34 wherein the method further comprises generating a computer program operable when stored in a memory associated with a said micro-controller to cause said micro-controller to store register values corresponding to said identified set of register values in registers in said micro-controller.

45. A computer readable medium in accordance with claim 34 wherein the method further comprises outputting data defining a schematic representation of a said micro-controller including labels identifying said set of signals required for routing to and from said selected peripherals to enable each of said peripherals to function.

46. A computer readable medium in accordance with claim 45 wherein said labels are grouped in said representation by the peripherals associated with said identified signals.

47. A computer readable medium in accordance with claim 45 wherein said labels are grouped in said representation so as to represent the physical locations of pins transferring said signals.

48. A computer readable medium in accordance with claim 34 comprising a disk.

49. A disk in accordance with claim 48 comprising a magnetic, magneto-optic or optical disk.

50. A computer readable medium in accordance with claim 34 comprising an electrical signal within a computer network.

51. A method of programming a micro-controller including a plurality of peripherals, the micro-controller being operable to route different selections of signals to and from said peripherals via a set of ports on the basis of register values stored within registers within said micro-controller, the method comprising:

generating a computer program in accordance with the method of claim 28;

storing said generated computer program in a memory associated with a micro-controller; and

running said stored program to cause said micro-controller to store register values corresponding to said identified set of register values in said registers in said micro-controller.

52. A micro-controller comprising:

a plurality of peripherals;

a plurality of selectors, operable to route signals to and from peripherals via a set of ports on the basis of register values stored in registers associated with said ports;

a memory operable to store a configuration program; and

a processing unit operable to process a configuration program stored in said memory to store register values in said registers; wherein each of said ports of said micro-controller comprises a port operable to transfer a single signal selected on the basis of a stored register value in the register associated with said port.

53. An apparatus in accordance with claim 1, wherein said first user input interface is operable to enable a user to input data identifying a selection of external peripherals external to a micro-controller and to utilise said selection of external peripherals to identify a selection of peripherals from the plurality of peripherals included in said micro-controller required to interface with said selection of external peripherals.

54. An apparatus in accordance with claim 53, wherein said first user input interface is operable to determine whether the selection of peripherals necessary to interface with an input selection of external peripherals require one or more peripherals not included in a micro-controller wherein said output module is operable to output data to cause the generation of a computer program for causing the micro-controller to emulate one or more additional peripherals in addition to peripherals included in said micro-controller.

55. An apparatus in accordance with claim 53, wherein said one or more external peripherals include registers operable to store settings data, said apparatus further comprising a user input interface operable to enable a user to input data identifying register values for storage in said one or more registers associated with said selection of external peripherals.

56. An apparatus in accordance with claim 55, wherein said further user interface is operable to associate each selection of external peripherals with an internal peripheral and to cause the output module to output data identifying a set of register values for the external peripherals and associated internal peripherals which are compatible with one another.

57. A method in accordance with claim 18, wherein said receiving data identifying a selection of peripherals comprises data identifying one or more peripherals external to a micro-controller, said method further comprising processing said received data to identify a selection of peripherals from the plurality of peripherals included in said micro-controller required to interface with said external peripherals.

58. A method in accordance with claim 57, wherein if said determination of peripherals necessary to interface with external peripherals includes one or more peripherals not included in said micro-controller wherein said output step comprises outputting data to

cause the generation of a computer program for causing the micro-controller to emulate one or more additional peripherals in addition to peripherals included in said micro-controller.

59. A method in accordance with claim 57, further receiving data identifying register values for storage in said one or more registers associated with said selection of external peripherals, wherein said output step includes identifying register values for storage within registers of one or more external peripherals corresponding to said received data.

60. An apparatus in accordance with claim 59, further comprising associating an external peripheral with an internal peripheral, wherein said output step comprises outputting data identifying a set of register values for the external peripherals and associated internal peripherals which are compatible with one another.

61. A computer readable medium in accordance with claim 34 wherein said receiving data identifying a selection of peripherals comprises data identifying one or more peripherals external to a micro-controller, said method further comprising processing said received data to identify a selection of peripherals from the plurality of peripherals included in said micro-controller required to interface with said external peripherals.

62. A computer readable medium in accordance with claim 61, wherein if said determination of peripherals necessary to interface with external peripherals includes one or more peripherals not included in said micro-controller wherein said output step comprises outputting data to cause the generation of a computer program for causing the micro-controller to emulate one or more additional peripherals in addition to peripherals included in said micro-controller.

63. A computer readable medium in accordance with claim 62, further receiving data identifying register values for storage in said one or more registers associated with said selection of external peripherals, wherein said output step includes identifying register values for storage within registers of one or more external peripherals corresponding to said received data.

64. A computer readable medium in accordance with claim 63, further comprising associating an external peripheral with an internal peripheral, wherein said output step

comprises outputting data identifying a set of register values for the external peripherals and associated internal peripherals which are compatible with one another.